

How to: Generate documents in Microsoft Flow

Introduction: Microsoft Flow is becoming more powerful every day and potentially is about to replace the long-standing classic Dynamics 365 workflows. In this article, we will present & showcase how DocumentsCorePack allows us to generate and process documents within Microsoft Flow.

How it Works: When using DocumentsCorePack every document that has to be generated is defined by a record of an entity called "AutoMergeWorkingItem". This record holds all mandatory definitions to generate the document:

1. **Record to use:** What record should the document be generated for (e.g. account, quote, etc.)
2. **Template to Execute:** Defines the DocumentsCorePack template that should be used to generate the document.
3. **Output Format:** The Filetype the document should be generated and returned as Pdf, docx, Html.
4. **(Optional) processing settings:** This is not a necessity but DocumentsCorePack allows to not just generate a document but also process the document by definitions on the AutoMerkeWorkingItem record (e.g. Print, Save to SharePoint, Send for e-Signing, etc.)

Once Such an AutoMergeWorkingItem is created, the DocumentsCorePack-service will generate the document and deliver it back as an attachment to this record.

It becomes clear that in order to generate a document within Flow the only thing that needs to be done is to "Create an AutoMergeWorking Item Record".

Note: The mandatory parameters required can be brought into flow in various ways. E.g. you could build a Canvas app that passes the parameters to the Flow or you can simply use the Common Data Service or Dynamics 365 connectors within Flow to load them (see Figure 1).

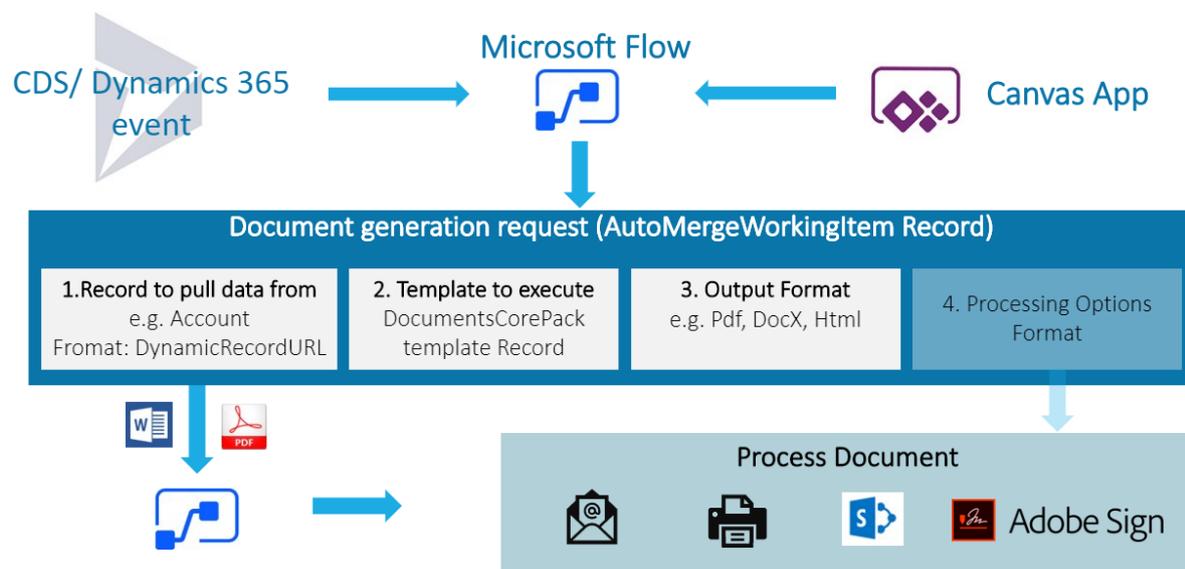


Figure 1: Document Generation request schema of DocumentsCorePack

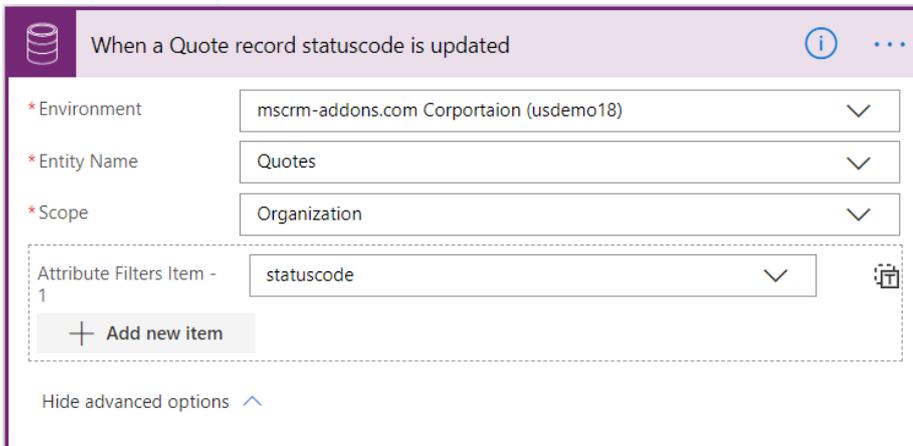
Step-by-Step: Create an AutoMergeWorkingItem in Microsoft Flow

Now that we know what is needed: Let's take a look at how such a document generation request has to look like.

Our use case is to generate a quote document in PDF – format upon activation of the quote.

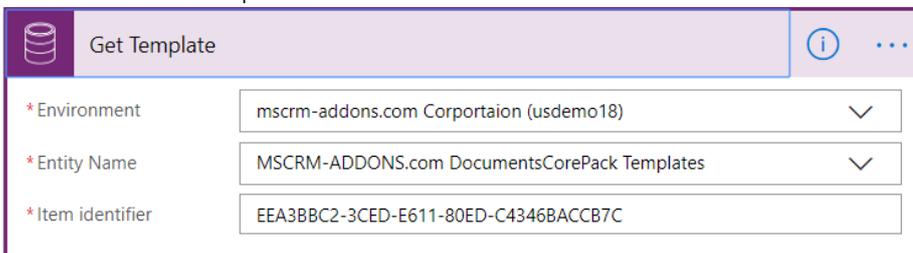
Note: In our scenario, we are using the common data service connector. Especially with the October 2019 update for the “Common Data Service (current environment)” it became a lot easier to do certain things like creating records. I want to emphasize though that it's of course also possible to use the Dynamics 365 connector.

- 1) **The Trigger of the flow:** In our scenario, the flow is defined to trigger a change of the status field of the quote entity as follows:



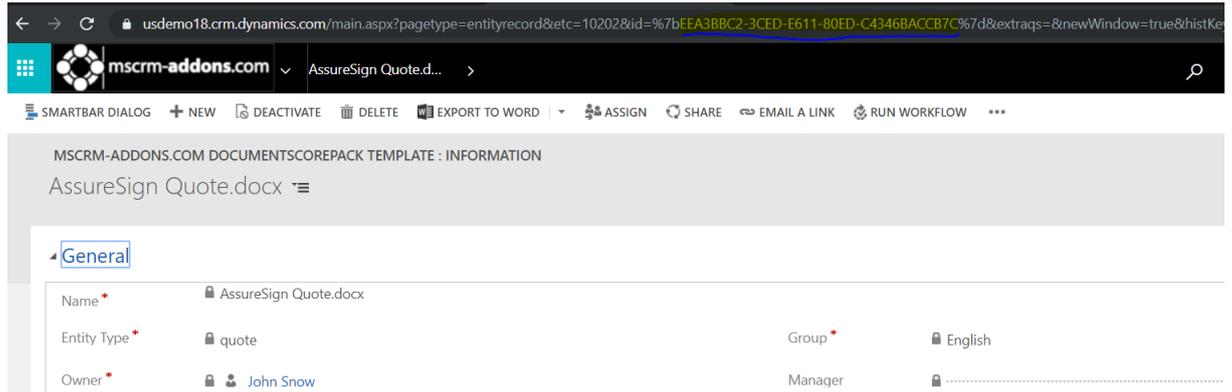
What's important: We will need the context of this step as it defines the record our document needs to pull the data from. So we have the first required parameter: The record to use!

- 2) **Get Template to Execute:** Here we use the Common Data Service- Get Record action to load our template. Since we always want to use the same quote-template we make it simple and “hardcode” our template.



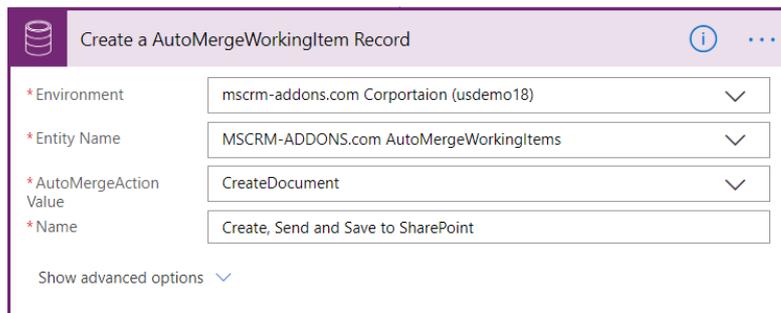
So what do we have here? Where does this GUID in the item identifier come from? To answer this we have to know that every template designed with the DocumentsCorePack template designer is stored as a record of the “mscrm-addons.com DocumentsCorePack Template” entity. As can be seen on the screenshot this is the entity we are getting a record of.

You can e.g. open the template record you want to use and copy the GUID from the URL



To Summarize: this Get Record step will provide us with the second parameter: The template to use! As we also know that our output format should be PDF we are all set to finally define our document generation request!

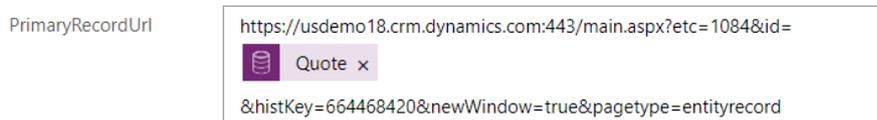
- 3) **Create AutoMergeWorking Item:** As all parameters are now clear we can use the Common Data Service – Create Record action to create our document generation request as follows:



We create a record of the Entity “MSCRM-ADDONS.COM AutoMergeWorkingItem” and choose the Action to be “Create Document”. We can also give it a name that is quite handy to maybe later identify and review the documents generated via this Flow.

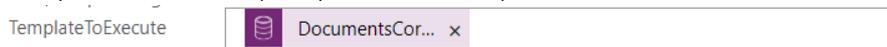
Now we need to set our parameters:

- a. Record to pull data from (PrimaryRecordUrl):



This is kind of challenging at this point. We did copy the URL of a quote we opened in Dynamics and replaced it with the “Quote”-parameter (Unique identifier of the quote from our Flow-Trigger action)

- b. Template to Execute (TemplateToExecute):



Here we reference the Unique Identifier parameter of our “Get Record” step where we did load the template above.

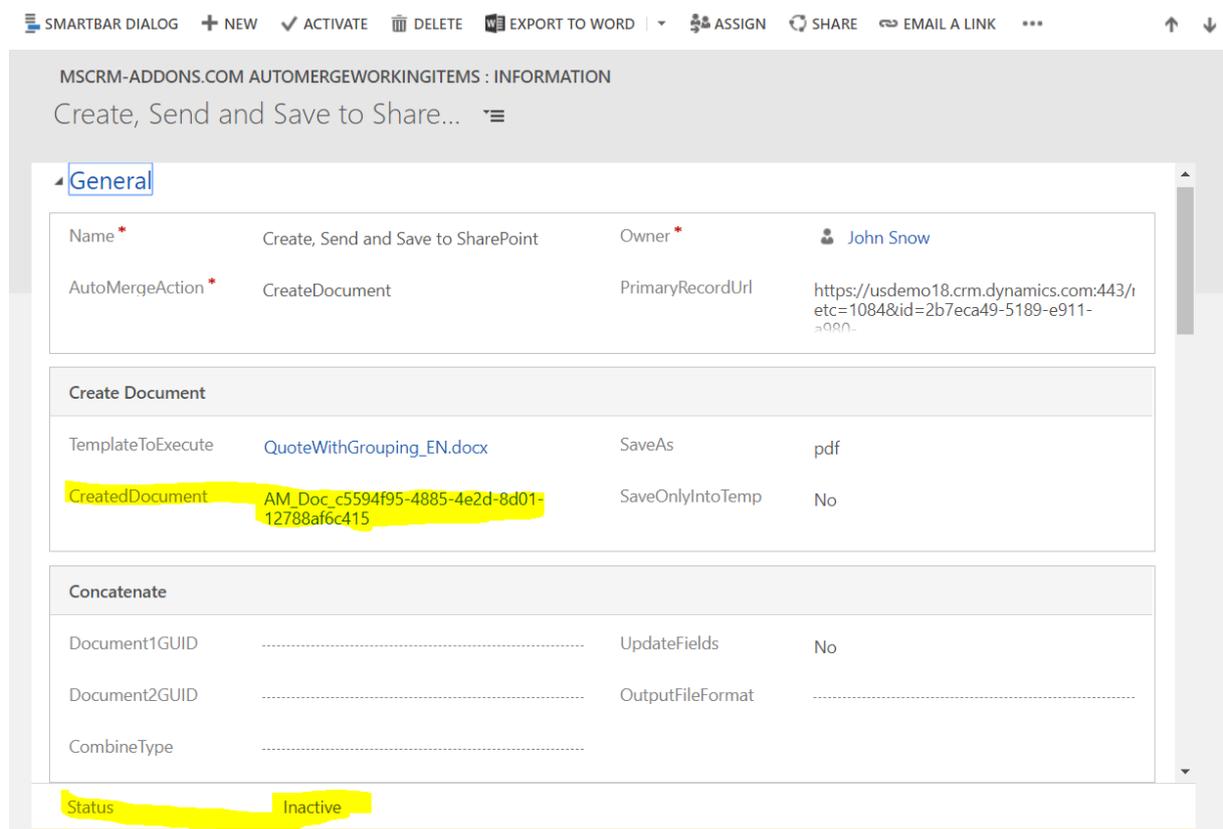
- c. FileType (SaveAs Value): This is a simple one – we simply select “PDF” from our picklist.

SaveAs Value

And that’s it! We did it. So every time a quote gets activated we will now get a document generated.

Next steps: Naturally you want to do something with the document once generated. As outlined above you can either use the AutoMergeWorkingItem to process the document but you can also process the document within your flow. The generated document can be found on the AutoMergeWorkingItem record -> Created Document section.

Note: This field will be filled when DocumentsCorePack successfully created the document. You will have to wait for the AutoMergeWorkingItem record to become “inactive”. This tells you that the document is ready.



SMARTBAR DIALOG + NEW ✓ ACTIVATE 🗑 DELETE 📄 EXPORT TO WORD | ▾ 👤 ASSIGN 🔄 SHARE 📧 EMAIL A LINK ... ⬆ ⬇

MSCRM-ADDONS.COM AUTOMERGEWORKINGITEMS : INFORMATION
Create, Send and Save to Share... ☰

General

Name *	Create, Send and Save to SharePoint	Owner *	John Snow
AutoMergeAction *	CreateDocument	PrimaryRecordUrl	https://usdemo18.crm.dynamics.com:443/etc=1084&tid=2b7eca49-5189-e911-a000-

Create Document

TemplateToExecute	QuoteWithGrouping_EN.docx	SaveAs	pdf
CreatedDocument	AM_Doc_c5594f95-4885-4e2d-8d01-12788af6c415	SaveOnlyIntoTemp	No

Concatenate

Document1GUID	UpdateFields	No
Document2GUID	OutputFileFormat
CombineType		

Status **Inactive**

We will soon provide you with another blog article + video on this topic.

Hope this was helpful – Let us know if you have any questions around this.

Cheers, Michael